Procedia Computer Science

Volume 80, 2016, Pages 734–743

ICCS 2016. The International Conference on Computational Science

Reduced Order Models for Pricing American Options under Stochastic Volatility and Jump-Diffusion Models

Maciej Balajewicz[1] and Jari Toivanen[2,3]

[1] University of Illinois at Urbana-Champaign, Champaign, Illinois, U.S.A.
mbalajew@illinois.edu
[2] Stanford University, Stanford, California, U.S.A.
toivanen@stanford.edu
[3] University of Jyväskylä, Jyväskylä, Finland
jari.toivanen@jyu.fi

Abstract

American options can be priced by solving linear complementary problems (LCPs) with parabolic partial(-integro) differential operators under stochastic volatility and jump-diffusion models like Heston, Merton, and Bates models. These operators are discretized using finite difference methods leading to a so-called full order model (FOM). Here reduced order models (ROMs) are derived employing proper orthogonal decomposition (POD) and non negative matrix factorization (NNMF) in order to make pricing much faster within a given model parameter variation range. The numerical experiments demonstrate orders of magnitude faster pricing with ROMs.

Keywords: reduced order model, option pricing, American option, linear complementary problem

1 Introduction

European options can be exercised only at expiry while American options can be exercised anytime until expiry. Due to this additional flexibility the American options can be more valuable and in order to avoid arbitrage the value must be always at least the same as the final payoff function. The seminal paper [6] by Black and Scholes models the value of underlying asset as a geometrical Brownian motion with a constant volatility. Since then it has become evident that more generic models are required for this value. Merton proposed adding log-normally distributed jumps to this model [19]. Heston [12] made the volatility to be a mean reverting stochastic process. Bates [4] combined the Heston stochastic volatility model and Merton jump-diffusion model.

The European options can be priced by solving a parabolic partial(-integro) differential equation. The American options lead to LCPs with the same operator. These operators are two dimensional with a stochastic volatility and one dimensional otherwise. The potential integral

 Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2016

doi:10.1016/j.procs.2016.05.360

part of the model results from the jumps. For American options finite difference discretizations lead to solving a discrete LCP at each time step. Efficient numerical methods for American options have been considered in [11, 13] for stochastic volatility models and in [9, 17, 25, 26, 28] for jump-diffusion models. Methods for the combined Bates model have been developed in [3, 27, 28, 31].

Unfortunately, high-fidelity simulations are still too expensive for many practical applications and reduced order modeling (ROM) is a promising tool for significantly alleviating computational costs [1, 5]. Most existing ROM approaches are based on projection. In projection-based reduced order modeling the state variables are approximated in a low-dimensional subspace. Basis for this subspace are typically constructed by Proper Orthogonal Decomposition (POD) [29] of a set of high-fidelity solution snapshots. In the case where the governing equations include a constraint equation it is often beneficial to construct a basis that satisfies these constraints a priori [7]. For example, in the case of non-negativity constraints, a non-negative basis can be constructed via non-negative matrix factorization (NNMF) [2]. Finally, projecting the high-fidelity model using a Galerkin or Petrov-Galerkin projection yields ROMs capable of delivering new solutions at a fraction of the computational costs of the high-fidelity model.

For pricing European options ROMs have been developed in [8, 23]. Only recently ROMs have been applied for pricing American options in [7, 10].

A common problem associated to option pricing is the calibration of model parameters to correspond to the market prices of options. This is typically formulated as a least squares -type optimization problem. The calibration is computationally expensive as it requires pricing a large number of options with varying parameters. The use of ROMs to reduce this computational cost has been studied in [20, 22, 21].

2 Full Order Models

Under the Merton model the price u of a European option can be obtained by solving the one-dimensional PIDE

$$\frac{\partial u}{\partial \tau} = \frac{1}{2}\sigma_s^2 s^2 \frac{\partial^2 u}{\partial s^2} + (r - \mu\xi) s \frac{\partial u}{\partial s} - (r+\mu)u + \mu \int_0^\infty u(sy,\tau)p(y)dy =: L^M u, \tag{1}$$

where $s \geq 0$ is the value of the underlying asset, $\tau = T - t$ is the time to expiry, σ_s is the volatility, r is the interest rate, μ is the jump intensity, and p is the log-normal jump distribution

$$p(y) = \frac{1}{y\delta\sqrt{2\pi}} \exp\left(-\frac{(\log y - \gamma)^2}{2\delta^2}\right). \tag{2}$$

The relative expected jump is $\xi = \exp\left(\gamma + \frac{1}{2}\delta^2\right) - 1$. The Black–Scholes model is obtained by setting the jump intensity μ to zero. Under the Bates model the price u of a European option can be obtained by solving the two-dimensional PIDE

$$\begin{aligned} \frac{\partial u}{\partial \tau} = {} & \frac{1}{2} v s^2 \frac{\partial^2 u}{\partial s^2} + \rho\sigma_v v s \frac{\partial^2 u}{\partial s \partial v} + \frac{1}{2}\sigma^2 v \frac{\partial^2 u}{\partial v^2} + (r - \mu\xi) s \frac{\partial u}{\partial s} + \kappa(\theta - v)\frac{\partial u}{\partial v} \\ & - (r+\mu)u + \mu \int_0^\infty u(sy, v, \tau)p(y)dy =: L^B u, \end{aligned} \tag{3}$$

where $v \geq 0$ is the variance, θ is its mean level, κ is the rate of mean reversion, σ_v is the volatility of variance, and ρ is the correlation between the asset value and the variance. The Heston model is obtained by setting the jump intensity μ to zero.

In the following, put options are considered. Their value at the expiry is given by the pay-off function $g(s) = \max\{K - s, 0\}$. Thus, the initial condition reads

$$u(s) = g(s) \qquad \text{and} \qquad u(s, v) = g(s) \tag{4}$$

for one-dimensional and two-dimensional models, respectively.

Due early exercise possibility the price u of an American option satisfies the LCP

$$\frac{\partial u}{\partial \tau} - Lu = \lambda, \quad u \geq g, \quad \lambda \geq 0, \quad \lambda(u - g) = 0, \tag{5}$$

where the operator L is either L^M or L^B depending on the model and λ is a Lagrange multiplier; see [14], for example. For computing an approximate solution the infinite domain is truncated at $s = s_{\max}$ and $v = v_{\max}$, where $s_{\max}$ and $v_{\max}$ are sufficiently large so that the error due to truncation is negligible. Dirichlet boundary conditions for American put options are

$$u = K \ \text{ at } \ s = 0 \qquad \text{and} \qquad u = 0 \ \text{ at } \ s = s_{\max}, \tag{6}$$

on the left and right boundaries, respectively. Under the stochastic volatility models, the Neumann boundary condition $\frac{\partial u}{\partial v} = 0$ is posed at $v = v_{\max}$. At $v = 0$ it is not necessary to pose a boundary condition. For the numerical solution, the LCP is formulated for $w = u - g$ instead of u as the positivity constraint $w \geq 0$ is more easily treated. This leads to the LCP

$$\frac{\partial w}{\partial \tau} - Lw = \lambda + Lg, \quad w \geq 0, \quad \lambda \geq 0, \quad \lambda w = 0. \tag{7}$$

A quadratic penalty formulation is obtained by choosing the Lagrange multiplier to be

$$\lambda = -\tfrac{1}{\varepsilon} \max\{-w, 0\}\, w. \tag{8}$$

This leads to the nonlinear P(I)DE

$$\frac{\partial w}{\partial \tau} - Lw + \tfrac{1}{\varepsilon} \max\{-w, 0\}\, w = Lg. \tag{9}$$

For the interval $[0, s_{\max}]$, a grid is defined by s_i, $i = 0, 1, 2, \ldots, N_s$. The spatial partial derivatives with respect to s are discretized using central finite difference

$$\frac{\partial w}{\partial s}(s_i) \approx \tfrac{1}{\Delta s_{i-1} + \Delta s_i} \left[-\tfrac{\Delta s_i}{\Delta s_{i-1}} u_{i-1} + \left(\tfrac{\Delta s_i}{\Delta s_{i-1}} - \tfrac{\Delta s_{i-1}}{\Delta s_i} \right) u_i + \tfrac{\Delta s_{i-1}}{\Delta s_i} u_{i+1} \right] \tag{10}$$

and

$$\frac{\partial^2 w}{\partial s^2}(s_i) \approx \tfrac{2}{\Delta s_{i-1} + \Delta s_i} \left[\tfrac{1}{\Delta s_{i-1}} u_{i-1} - \left(\tfrac{1}{\Delta s_{i-1}} + \tfrac{1}{\Delta s_i} \right) u_i + \tfrac{1}{\Delta s_i} u_{i+1} \right], \tag{11}$$

where $\Delta s_i = s_{i+1} - s_i$. Similarly for the interval $[0, v_{\max}]$, a grid is defined v_j, $j = 0, 1, 2, \ldots, N_v$. The spatial partial derivatives with respect to v are discretized using the above central finite difference. A nine-point finite difference stencil for $\frac{\partial^2 w}{\partial s \partial v}$ is obtained by employing the central finite differences in both directions. The integrals can be discretized using a second-order accurate quadrature formula. Here the linear interpolation is used for w between grid points and exact integration; see [24], for details. Under the Merton model the discretization of the integral leads to a full matrix while under the Bates model it leads to full diagonal blocks.

Under models without jumps the time discretization is performed by taking the first time steps using the implicit Euler method and after using the second-order accurate BDF2 method. Under jump models the integral is treated explicitly. In the first time step using the explicit Euler method and in the following time steps using the linear extrapolation based on the two

previous time steps. This IMEX-BDF2 method is described in [26]. With the explicit treatment of the integral it is not necessary to solve systems with dense matrices. At the time $(k+1)\Delta\tau$, the grid point values contained in the vector $\mathbf{w}^{k+1}$ are obtained by solving the system

$$\begin{aligned}&\left(\mathbf{I}+\tfrac{2}{3}\Delta\tau\mathbf{D}+\tfrac{1}{\varepsilon}\operatorname{diag}\left(\max\left\{-\mathbf{w}^{k+1},0\right\}\right)\right)\mathbf{w}^{k+1}\\&=\left(\tfrac{4}{3}\mathbf{w}^{k}-\tfrac{1}{3}\mathbf{w}^{k-1}\right)+\Delta\tau\mathbf{J}\left(\tfrac{4}{3}\mathbf{w}^{k}-\tfrac{2}{3}\mathbf{w}^{k-1}\right)+\tfrac{2}{3}\Delta\tau\mathbf{f},\end{aligned}\tag{12}$$

where the matrices $\mathbf{J}$ and $\mathbf{D}$ corresponds to the terms due to the jumps and the rest, respectively. The vector $\mathbf{f}$ contains the grid point values of Lg. The operator $\operatorname{diag}(\cdot)$ gives a diagonal matrix with the diagonal entries defined by the argument vector. The maximum is taken componentwise. The system (12) can be expressed more compactly as

$$\left(\mathbf{A}+\tfrac{1}{\varepsilon}\operatorname{diag}\left(\max\left\{-\mathbf{w}^{k+1},0\right\}\right)\right)\mathbf{w}^{k+1}=\mathbf{r}^{k+1}\tag{13}$$

with suitably defined $\mathbf{A}$ and $\mathbf{r}^{k+1}$. The discrete counterpart of the Lagrange multiplier λ in (8) reads

$$\boldsymbol{\lambda}^{k+1}=-\tfrac{1}{\varepsilon}\operatorname{diag}\left(\max\left\{-\mathbf{w}^{k+1},0\right\}\right)\mathbf{w}^{k+1}.\tag{14}$$

3 Reduced Order Models

Let $\mathbf{U}\in\mathbb{R}^{N\times n}$ be basis for $\mathbf{w}$. The reduced solution $\mathbf{w}=\mathbf{U}\mathbf{w}_r$ is governed by

$$\left(\mathbf{U}^T\mathbf{A}\mathbf{U}+\tfrac{1}{\varepsilon}\mathbf{U}^T\operatorname{diag}\left(\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}\right)\mathbf{U}\right)\mathbf{w}_r^{k+1}=\mathbf{U}^T\mathbf{r}^{k+1}.\tag{15}$$

The product $\mathbf{U}^T\operatorname{diag}\left(\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}\right)\mathbf{U}$ is the only product in (15) that cannot be precomputed offline. Since the cost of evaluating this product scales with the size of the full-order model, Eq. (15) does not offer major computational savings.

Let $\mathbf{U}_\lambda\in\mathbb{R}^{N\times n_\lambda}$ be basis for $\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}$. The basis $\mathbf{U}_\lambda$ is assumed to be non negative, that is, all components of $\mathbf{U}_\lambda$ are non negative. Thus, we assume

$$\mathbf{U}_\lambda\mathbf{h}_r\approx\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}.\tag{16}$$

Performing a Galerkin projection yields

$$\mathbf{U}_\lambda^T\mathbf{U}_\lambda\mathbf{h}_r\approx\mathbf{U}_\lambda^T\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}.\tag{17}$$

Unfortunately, Equation (17) still does not yield a computationally efficient model because $\mathbf{U}_\lambda^T\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}\neq\max\left\{-\mathbf{U}_\lambda^T\mathbf{U}\mathbf{w}_r^{k+1},0\right\}$ can not be precomputed offline.

To achieve computational speed-up, a Petrov-Galerkin projection is performed using a binary selection matrix.

Consider a binary matrix $\mathbf{P}\in\{0,1\}^{N\times m}$ whose columns correspond to the i_j column of the identify matrix, $\mathbf{P}_{:,i_j}=\mathbf{I}_{:,i_j}$ and where the index i_j corresponds to the location of the maximum component of the j^{th} column of $\mathbf{U}_\lambda$. Since $\mathbf{U}_\lambda>\mathbf{0}$ and $\mathbf{h}_r>\mathbf{0}$, the full solution $\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}$ is positive if and only if $\mathbf{P}^T\mathbf{U}_\lambda\mathbf{h}_r>\mathbf{0}$. With the assumption outlined in (16), it follows that

$$\mathbf{P}^T\mathbf{U}_\lambda\mathbf{h}_r\approx\mathbf{P}^T\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}.\tag{18}$$

and finally

$$\mathbf{P}^T\max\left\{-\mathbf{U}\mathbf{w}_r^{k+1},0\right\}=\max\left\{-\mathbf{P}^T\mathbf{U}\mathbf{w}_r^{k+1},0\right\}.\tag{19}$$

Thus, the final form of the ROM is as follows

$$\left(\mathbf{A}_r + \tfrac{1}{\varepsilon}\mathbf{C}^T \operatorname{diag}\left(\max\left\{-\mathbf{C}\mathbf{w}_r^{k+1}, 0\right\}\right)\mathbf{C}\right)\mathbf{w}_r^{k+1} = \mathbf{r}_r^{k+1}, \tag{20}$$

where $\mathbf{A}_r = \mathbf{U}^T\mathbf{A}\mathbf{U}$, $\mathbf{C} = \mathbf{P}^T\mathbf{U}$, and $\mathbf{r}_r = \mathbf{U}^T\mathbf{r}$. All components in Equation (20) scale with the size of the reduced-order model.

The basis $\mathbf{U} \in \mathbb{R}^{N\times n}$, and $\mathbf{U}_\lambda \in \mathbb{R}^{N\times n_\lambda}$ are constructed by solving a low-rank approximation problem for a set of solution snapshot. A solution snapshot, or simply a snapshot, is defined as a state vector $\mathbf{w}^k$ computed as the solution of (12) for some instance of its parameters. A solution matrix is defined as a matrix whose columns are individual snapshots.

To construct $\mathbf{U} \in \mathbb{R}^{N\times n}$, the following optimization problem is solved

$$\underset{\mathbf{U}\in\mathbb{R}^{N\times n},\,\mathbf{V}\in\mathbb{R}^{n\times K}}{\text{minimize}} \|\mathbf{X} - \mathbf{U}\mathbf{V}\|_F^2, \tag{21}$$

where K is the number of solution snapshots. Hence, the basis $\mathbf{U}$ is comprised of the first n left singular vectors of the snapshot matrix $\mathbf{X}$ and $\mathbf{V} = \mathbf{\Sigma}\mathbf{W}^T$, where $\mathbf{\Sigma}$ is the diagonal matrix of the first n singular values of $\mathbf{\Sigma}$, and $\mathbf{W}$ is the matrix of its first n right singular vectors.

To construct $\mathbf{U}_\lambda \in \mathbb{R}^{N\times n_\lambda}$, the following constrained optimization problem is solved

$$\begin{aligned} \underset{\mathbf{U}_\lambda\in\mathbb{R}^{N\times n_\lambda},\,\mathbf{V}_\lambda\in\mathbb{R}^{n_\lambda\times K}}{\text{minimize}} \quad & \|\mathbf{X}_\lambda - \mathbf{U}_\lambda\mathbf{V}_\lambda\|_F^2, \\ \text{subject to} \quad & \mathbf{U}_\lambda \geq \mathbf{0} \\ & \mathbf{V}_\lambda \geq \mathbf{0} \end{aligned} \tag{22}$$

where $\mathbf{X}_\lambda \in \mathbb{R}^{N\times K}$ is the snapshot matrix comprised of the projected constraint violation $\max\left\{-\mathbf{U}^T\mathbf{U}\mathbf{w}^k, 0\right\}$. Unlike problem (21), problem (22) does not have a closed form solution. Consequently, this problem is usually solved using an iterative method that typically converges to a local minimum. Examples of such methods are the original multiplicative updating rule [18], the alternating non-negativity least-squares method [15], and block coordinate descent algorithms [16].

4 Numerical Experiments

All numerical examples price an American put option with the strike price $K = 100$ and the expiry $T = 0.5$. Only at the money option is considered, that is, the value of u at $s = K$ is sought. Under the stochastic volatility models the value of u is computed at the variance $v = \theta$. The full order models are discretized using quadratically refined spatial grids similar to ones employed by the FD-NU method in [30]. The s-grid is defined by $s_i = \left[\left(\frac{i}{\alpha N_s} - 1\right)\left|\frac{i}{\alpha N_s} - 1\right| + 1\right]K$, $i = 0, 2, \ldots, N_s$ with $\alpha = \frac{3}{8}$. For the stochastic volatility models the variance grid is defined by $v_j = \left(\frac{j}{N_v}\right)^2 v_{\max}$ with $v_{\max} = 1$. The uniform time steps are given by $\Delta\tau = \frac{1}{N_\tau}T$. In the experiments the number of spatial and temporal steps are chosen to be $N_s = 128$, $N_v = 64$, and $N_\tau = 32$. With this choice and the employed parameter ranges the absolute discretization error is about 10^{-2} or less.

The snapshot matrix $\mathbf{X}$ is given by all vectors $\mathbf{w}^k$, $k = 1, 2, \ldots, N_\tau$, in all training runs. For these training runs each model parameter is sampled at its extreme values and at the midpoint between them. Thus, with two, five, and eight model parameters there are $3^2 = 9$, $3^5 = 243$, and $3^8 = 6561$ training runs, respectively. In the predictive ROM simulations, each parameter

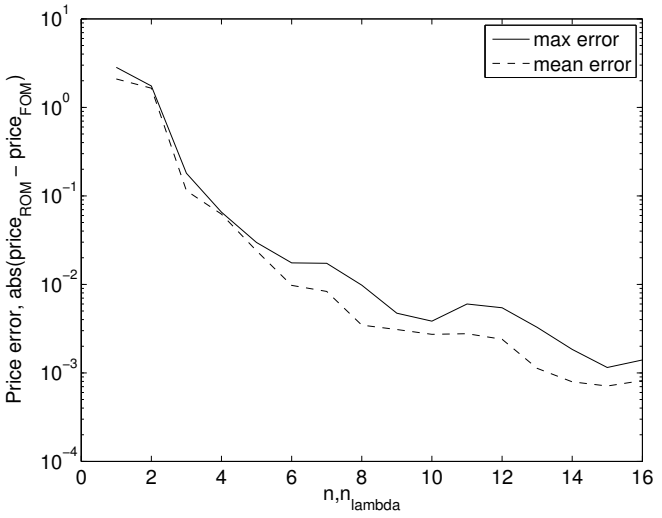

Figure 1: Under the Black–Scholes model the error with respect to the the reduced basis size $n = n_\lambda$

has two values which are the midpoint values between the values used in the training. Thus, with two, five, and eight model parameters there are $2^2 = 4$, $3^5 = 32$, and $2^8 = 256$ prediction runs, respectively. The sizes of the two reduced basis given by n and n_λ are chosen to be the same. The measured error is the absolute difference between the prices given by the reduced order model and the full order model.

4.1 Black–Scholes Model

The model parameters for the Black–Scholes model are varied in the range:

$$(r,\, \sigma_s) \in [0.025,\, 0.035] \times [0.35,\, 0.45]. \tag{23}$$

The value of the option varies roughly in the range $[9.06, 12.04]$. Figure 1 shows the reduction of the maximum and mean errors with the growth of the reduced basis sizes $n = n_\lambda$.

4.2 Merton Model

The model parameters for the Merton model are varied in the range:

$$(r,\, \sigma_s,\, \mu,\, \delta,\, \gamma) \in [0.025,\, 0.035] \times [0.35,\, 0.45] \times [0.15,\, 0.25] \times [0.3,\, 0.5] \times [-0.7,\, -0.3]. \tag{24}$$

The value of the option varies roughly in the range $[9.65, 14.08]$. Figure 2 shows the reduction of the maximum and mean errors with the growth of the reduced basis sizes $n = n_\lambda$.

4.3 Heston Model

The model parameters for the Heston model are varied in the range:

$$(r,\, \kappa,\, \theta,\, \sigma_v,\, \rho) \in [0.025,\, 0.035] \times [3,\, 5] \times [0.35^2,\, 0.45^2] \times [0.35,\, 0.45] \times [-0.75,\, -0.25]. \tag{25}$$

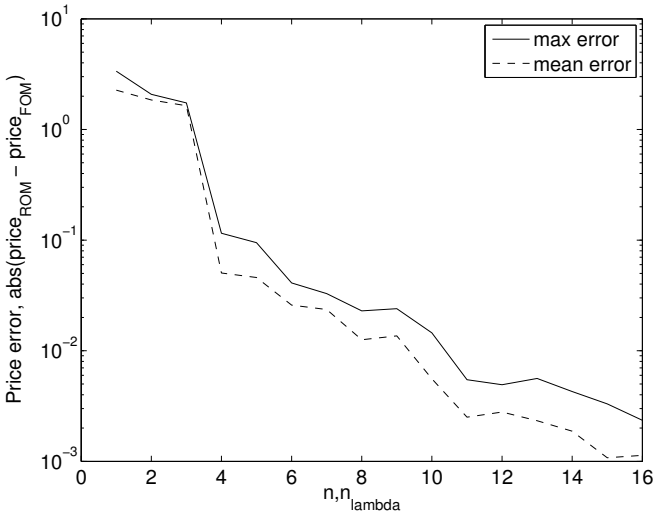

Figure 2: Under the Merton model the error with respect to the the reduced basis size $n = n_\lambda$

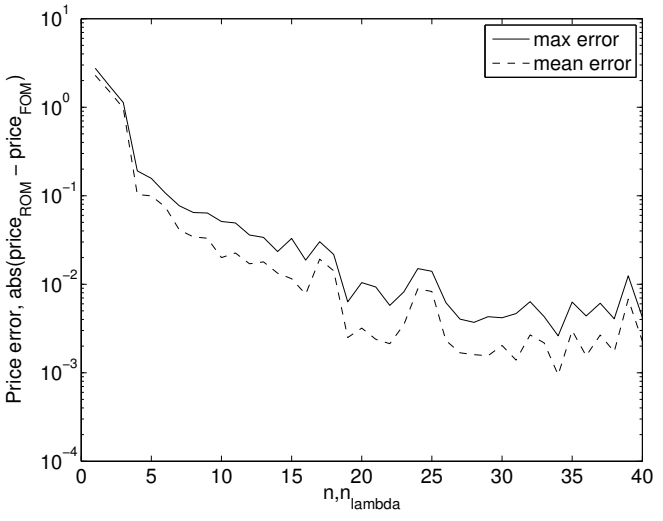

Figure 3: Under the Heston model the error with respect to the the reduced basis size $n = n_\lambda$

The value of the option varies roughly in the range $[8.87, 11.98]$. Figure 3 shows the reduction of the maximum and mean errors with the growth of the reduced basis sizes $n = n_\lambda$.

4.4 Bates Model

The model parameters for the Bates model are varied in the range:

$$\begin{aligned}(r,\, \kappa,\, \theta,\, \sigma_v,\, \rho,\, \mu,\, \delta,\, \gamma) \in\, & [0.025,\, 0.035] \times [3,\, 5] \times [0.35^2,\, 0.45^2] \times [0.35,\, 0.45] \times \\ & [-0.75,\, -0.25] \times [0.15,\, 0.25] \times [0.3,\, 0.5] \times [-0.7,\, -0.3].\end{aligned} \tag{26}$$

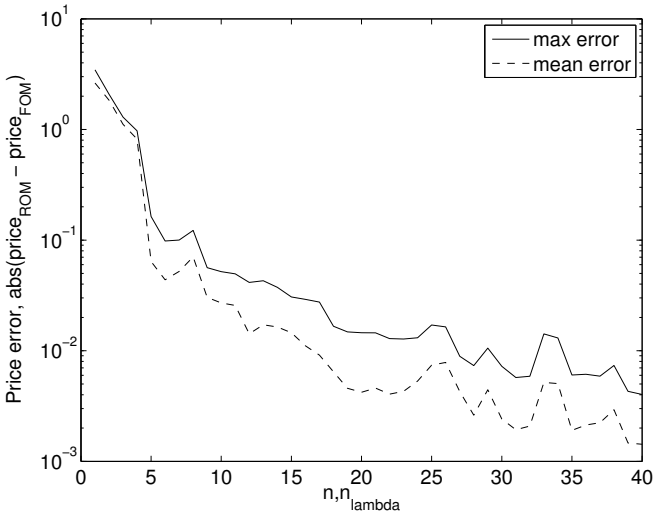

Figure 4: Under the Bates model the error with respect to the the reduced basis size $n = n_\lambda$

Table 1: CPU times in seconds for on-line computations.

	FOM		ROM		
Model	# DOF	CPU time	# DOF	CPU time	speed-up
Black–Scholes	127	0.037	16	0.034	1.1
Merton	127	0.039	16	0.034	1.1
Heston	8255	7.52	40	0.044	171
Bates	8255	7.42	40	0.044	169

The value of the option varies roughly in the range $[9.53, 14.07]$. Figure 4 shows the reduction of the maximum and mean errors with the growth of the reduced basis sizes $n = n_\lambda$.

4.5 Computational Speed-up

For each problem considered, the speed-up factor delivered by its ROM for the online computations is reported in Table 1. All models are solved in MATLAB on a Intel Xeon 2.6GHz CPU and all CPU times were measured using the `tic-toc` function on a single computational thread via the `-singleCompThread` start-up option. A ROM is integrated in time using the same scheme and time-step used to solve its corresponding FOM; see Section 2 for details. The online speed-up is calculated by evaluating the ratio between the time-integration of the FOM and the time-integration of the ROM.

5 Conclusions

Reduced order models (ROMs) were constructed for pricing American options under jump-diffusion and stochastic volatility models. They are based on a penalty formulation of the linear complementarity problem. The finite difference discretized differential operator and the penalty term are projected using basis resulting from a proper orthogonal decomposition and a non

negative matrix factorization. In numerical experiments, from two to eight model parameters are varied in a given range. For the one-dimensional Black–Scholes and Merton models about 16 ROM basis vectors were enough to reach 0.1% accuracy. For the two-dimensional Heston and Bates models about 40 basis vectors were needed to reach the same accuracy. For these two-dimensional models the computational speed-up was about 170 when the full order model (FOM) and ROM have roughly the same 0.1% accuracy level. The solution of the FOM and the ROM for two-dimensional models required about 7.5 and 0.044 seconds, respectively. With the one-dimensional models the speed-up was negligible. Particularly the results with the Bates model and eight parameters varying are impressive. A potential application for these ROMs is fast and accurate calibration of the model parameters based on market data.

References

[1] A.C. Antoulas. *Approximation of large-scale dynamical systems*, volume 6 of *Advances in Design and Control*. SIAM, 2005.

[2] M. Balajewicz, D. Amsallem, and C. Farhat. Projection-based model reduction for contact problems. *Int. J. Numer. Meth. Eng.*, 2015. Available at http://dx.doi.org/10.1002/nme.5135.

[3] L. V. Ballestra and C. Sgarra. The evaluation of American options in a stochastic volatility model with jumps: an efficient finite element approach. *Comput. Math. Appl.*, 60(6):1571–1590, 2010.

[4] D. S. Bates. Jumps and stochastic volatility: Exchange rate processes implicit Deutsche mark options. *Review Financial Stud.*, 9(1):69–107, 1996.

[5] P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015.

[6] F. Black and M. Scholes. The pricing of options and corporate liabilities. *J. Political Economy*, 81:637–654, 1973.

[7] O. Burkovska, B. Haasdonk, J. Salomon, and B. Wohlmuth. Reduced basis methods for pricing options with the Black-Scholes and Heston models. *SIAM J. Financial Math.*, 6(1):685–712, 2015.

[8] R. Cont, N. Lantos, and O. Pironneau. A reduced basis for option pricing. *SIAM J. Financial Math.*, 2(1):287–316, 2011.

[9] Y. d'Halluin, P. A. Forsyth, and G. Labahn. A penalty method for American options with jump diffusion processes. *Numer. Math.*, 97(2):321–352, 2004.

[10] B. Haasdonk, J. Salomon, and B. Wohlmuth. A reduced basis method for the simulation of American options. In *Numerical mathematics and advanced applications 2011*, pages 821–829. Springer, Heidelberg, 2013.

[11] T. Haentjens and K. J. in 't Hout. ADI schemes for pricing American options under the Heston model. *Appl. Math. Finance*, 22(3):207–237, 2015.

[12] S. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Rev. Financial Stud.*, 6:327–343, 1993.

[13] S. Ikonen and J. Toivanen. Efficient numerical methods for pricing American options under stochastic volatility. *Numer. Methods Partial Differential Equations*, 24(1):104–126, 2008.

[14] K. Ito and J. Toivanen. Lagrange multiplier approach with optimized finite difference stencils for pricing American options under stochastic volatility. *SIAM J. Sci. Comput.*, 31(4):2646–2664, 2009.

[15] H. Kim and H. Park. Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method. *SIAM Journal on Matrix Analysis and Applications*, 30(2):713–730, 2008.

[16] J. Kim, Y. He, and H. Park. Algorithms for nonnegative matrix and tensor factorizations: a unified view based on block coordinate descent framework. *Journal of Global Optimization*, 58(2):285–319,

2013.

[17] Y. Kwon and Y. Lee. A second-order tridiagonal method for American options under jump-diffusion models. *SIAM J. Sci. Comput.*, 33(4):1860–1872, 2011.

[18] D.D. Lee and H.S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[19] R. C. Merton. Option pricing when underlying stock returns are discontinuous. *J. Financial Econ.*, 3:125–144, 1976.

[20] O. Pironneau. Calibration of options on a reduced basis. *J. Comput. Appl. Math.*, 232(1):139–147, 2009.

[21] E. W. Sachs, M. Schneider, and M. Schu. Adaptive trust-region POD methods in PIDE-constrained optimization. In *Trends in PDE constrained optimization*, volume 165 of *Internat. Ser. Numer. Math.*, pages 327–342. Birkhäuser/Springer, Cham, 2014.

[22] E. W. Sachs and M. Schu. Reduced order models in PIDE constrained optimization. *Control Cybernet.*, 39(3):661–675, 2010.

[23] E. W. Sachs and M. Schu. *A priori* error estimates for reduced order models in finance. *ESAIM Math. Model. Numer. Anal.*, 47(2):449–469, 2013.

[24] S. Salmi and J. Toivanen. An iterative method for pricing American options under jump-diffusion models. *Appl. Numer. Math.*, 61(7):821–831, 2011.

[25] S. Salmi and J. Toivanen. A comparison and survey of finite difference methods for pricing American options under finite activity jump-diffusion models. *Int. J. Comput. Math.*, 89(9):1112–1134, 2012.

[26] S. Salmi and J. Toivanen. IMEX schemes for pricing options under jump-diffusion models. *Appl. Numer. Math.*, 84:33–45, 2014.

[27] S. Salmi, J. Toivanen, and L. von Sydow. Iterative methods for pricing American options under the Bates model. In *Proceedings of 2013 International Conference on Computational Science*, volume 18 of *Procedia Computer Science Series*, pages 1136–1144. Elsevier, 2013.

[28] S. Salmi, J. Toivanen, and L. von Sydow. An IMEX-scheme for pricing options under stochastic volatility models with jumps. *SIAM J. Sci. Comput.*, 36(5):B817–B834, 2014.

[29] L. Sirovich. Turbulence and the dynamics of coherent structures. Part I: coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987.

[30] L. von Sydow, L.J. Höök, E. Larsson, E. Lindström, S. Milovanović, J. Persson, V. Shcherbakov, Y. Shpolyanskiy, S. Sirén, J. Toivanen, J. Walden, M. Wiktorsson, J. Levesley, J. Li, C.W. Oosterlee, M.J. Ruijter, A. Toropov, and Y. Zhao. BENCHOP – The BENCHmarking project in option pricing. *Int. J. Comput. Math.*, 92(12):2361–2379, 2015.

[31] L. von Sydow, J. Toivanen, and C. Zhang. Adaptive finite differences and IMEX time-stepping to price options under Bates model. *Int. J. Comput. Math.*, 92(12):2515–2529, 2015.